

Algebraic multigrid and multilevel methods

A general introduction

Yvan Notay*

ynotay@ulb.ac.be

Université Libre de Bruxelles
Service de Métrologie Nucléaire

May 20, 2005, Leuven

* Supported by the "Fonds National de la Recherche Scientifique", Maître de recherches
<http://homepages.ulb.ac.be/~ynotay>

Algebraic multigrid and multilevel methods – p.1/66

Large sparse **discrete PDE** systems

$$A \mathbf{u} = \mathbf{b} .$$

- Iterative methods accelerated by **preconditioning**: easily invertible B such that $B \approx A$.
- **Multigrid & multilevel** methods: often very efficient.
- Basic principle (two-grid): obtain fast the convergence by solving a **smaller problem**, on a **coarser grid**.
- **Recursive use**: the coarse grid problem is solved using the same two-grid preconditioner.
- This seminar: emphasis on **algebraic** methods (that work using only the information in A).

Algebraic multigrid and multilevel methods – p.2/66

Algebraic methods: field of application

- Robust for scalar elliptic PDEs with standard discretization.
- Emphasis on (theory for) symmetric problems (self-adjoint PDEs), but work in unsymmetric cases as well (e.g. convection diffusion problems).
- Ongoing research for systems of PDEs (efficient preconditioning of each diagonal block).
- Does not work well for indefinite problems (some eigenvalues with negative real part); e.g.: Helmholtz.

Remark: **AMG** is the generic name of a family of methods, but also the specific name of Ruge & Stüben method.

Algebraic multigrid and multilevel methods – p.3/66

Outline

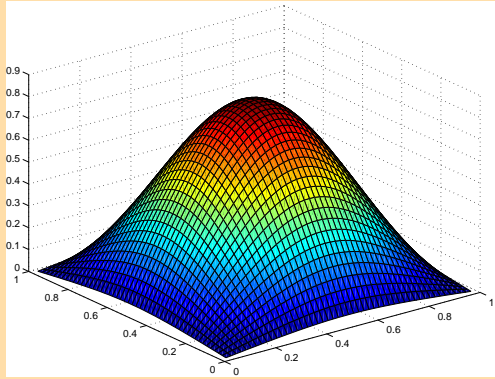
1. An introductory example.
2. Needed ingredients: algebraic coarsening and algebraic interpolation.
3. The different schemes and their algebraic properties.
4. Algebraic interpolation.
5. Algebraic coarsening: standard from AMG and aggregation.
6. Checking & correcting the coarsening.
7. From two- to multi-level: cycling strategies.
8. Some numerical illustrations

Algebraic multigrid and multilevel methods – p.4/66

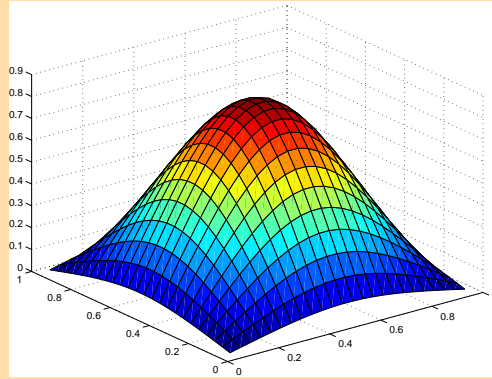
$$\text{PDE: } -\Delta u = 20 e^{-10((x-0.5)^2+(y-0.5)^2)} \text{ in } \Omega = (0,1) \times (0,1)$$

$$u = 0 \text{ on } \partial\Omega$$

Uniform grid with mesh size h , five-point finite difference.



Solution with $h^{-1} = 50$



Solution with $h^{-1} = 25$

Algebraic multigrid and multilevel methods – p.5/66

Fine grid (system to solve):

$$A \mathbf{u} = \mathbf{b} .$$

Coarse grid (auxiliary system):

$$A_C \mathbf{u}_C = \mathbf{b}_C .$$

\mathbf{u}_C may be computed and prolonged (by interpolation) on the fine grid:

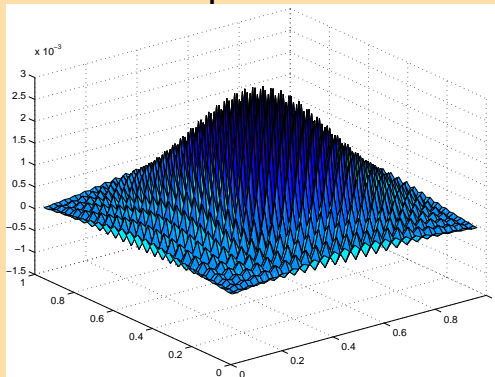
$$\mathbf{u}^{(1)} = p \mathbf{u}_C$$

$\mathbf{u}^{(1)}$ may serve as initial approximation, i.e., one solves

$$A(\mathbf{u}^{(1)} + \mathbf{x}) = \mathbf{b} \quad \text{or} \quad A \mathbf{x} = \mathbf{b} - A p A_C^{-1} \mathbf{b}_C .$$

Algebraic multigrid and multilevel methods – p.6/66

Error on the fine grid after interpolation



$$\frac{\|\mathbf{u} - \mathbf{u}^{(1)}\|}{\|\mathbf{u}\|} = 0.0019$$

Algebraic multigrid and multilevel methods – p.7/66

$$A(\mathbf{u}^{(1)} + \mathbf{x}) = \mathbf{b} \quad \text{or} \quad A \mathbf{x} = \mathbf{b} - A p A_C^{-1} \mathbf{b}_C = \mathbf{r}^{(1)} .$$

(1) Restrict on the coarse grid:

$$\mathbf{r}_C = r \mathbf{r}^{(1)} .$$

(2) Solve on the coarse grid:

$$\mathbf{x}_C^{(2)} = A_C^{-1} \mathbf{r}_C .$$

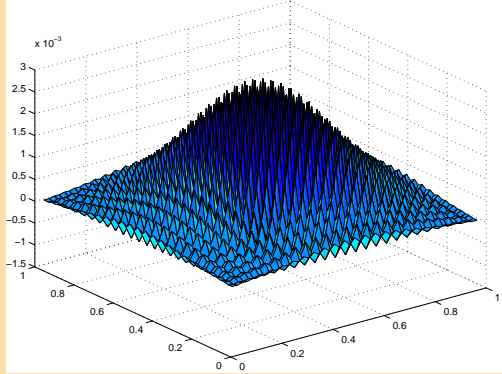
(3) Prolongate:

$$\mathbf{x}^{(2)} = p \mathbf{x}_C^{(2)} ,$$

$$\mathbf{u}^{(2)} = \mathbf{u}^{(1)} + \mathbf{x}^{(2)} .$$

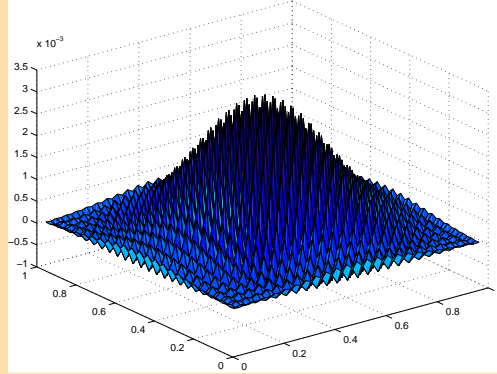
Algebraic multigrid and multilevel methods – p.8/66

Error on the fine grid after interpolation



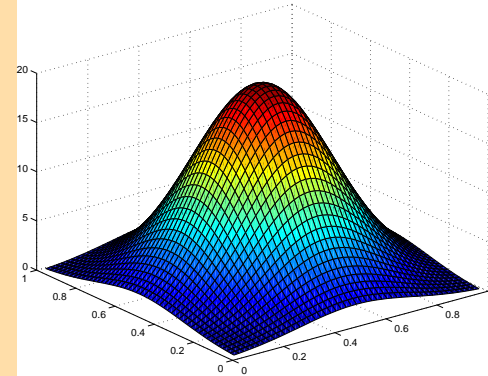
$$\frac{\|\mathbf{u} - \mathbf{u}^{(1)}\|}{\|\mathbf{u}\|} = 0.0019$$

Repeating the process ...

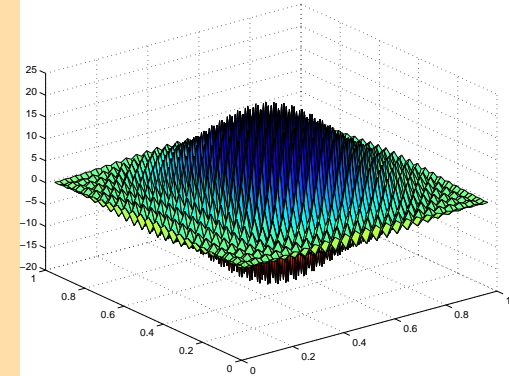


$$\frac{\|\mathbf{u} - \mathbf{u}^{(2)}\|}{\|\mathbf{u}\|} = 0.0018$$

Initial residual (r.h.s.)



After coarse grid correction



$$\frac{\|\mathbf{b} - A p A_C^{-1} \mathbf{b}_C\|}{\|\mathbf{b}\|} = 0.7142$$

Explanation

Assume (for simplicity) that $\mathbf{b}_C = r \mathbf{b}$.

One has

$$\begin{aligned} \mathbf{u} - \mathbf{u}^{(1)} &= \mathbf{u} - p A_C^{-1} r \mathbf{b} \\ &= (I - p A_C^{-1} r A) \mathbf{u}, \end{aligned}$$

$$\mathbf{u} - \mathbf{u}^{(2)} = (I - p A_C^{-1} r A)^2 \mathbf{u},$$

etc. Similarly

$$\begin{aligned} \mathbf{r}^{(1)} &= \mathbf{b} - A p A_C^{-1} r \mathbf{b} \\ &= (I - A p A_C^{-1} r) \mathbf{r}^{(0)}. \end{aligned}$$

$p A_C^{-1} r$ has rank $n_C \rightarrow$

$$\rho(I - A p A_C^{-1} r) = \rho(I - p A_C^{-1} r A) \geq 1.$$

Smoother enters the scene

$\mathbf{u} - \mathbf{u}^{(1)}$ and $\mathbf{r}^{(1)}$ very oscillatory

\rightarrow improve $\mathbf{u}^{(1)}$ with a simple iterative method, efficient in **smoothing** the error & residual.

Example: symmetric Gauss-Seidel (SGS)

$$L \mathbf{u}^{(1+1/2)} = \mathbf{b} - (A - L) \mathbf{u}^{(1)}, \quad (L = \text{low}(A))$$

$$U \mathbf{u}^{(2)} = \mathbf{b} - (A - U) \mathbf{u}^{(1+1/2)}. \quad (U = \text{upp}(A))$$

Same as

$$\mathbf{u}^{(2)} = \mathbf{u}^{(1)} + M^{-1} \mathbf{r}^{(1)}, \quad M = L D^{-1} U \quad (D = \text{diag}(A))$$

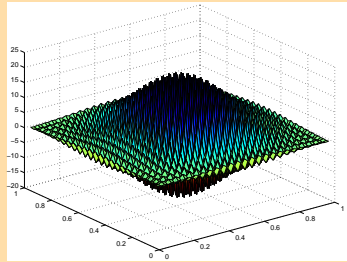
Thus:

$$\mathbf{u} - \mathbf{u}^{(2)} = (I - M^{-1} A) (\mathbf{u} - \mathbf{u}^{(1)})$$

$$\mathbf{r}^{(2)} = (I - A M^{-1}) \mathbf{r}^{(1)}$$

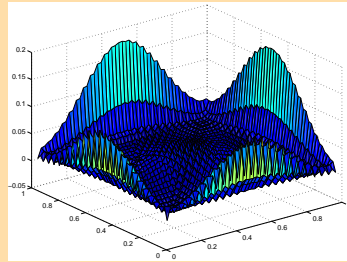
One may repeat: $\mathbf{r}^{(m+1)} = (I - A M^{-1})^m \mathbf{r}^{(1)}$.

Residual after CG correction



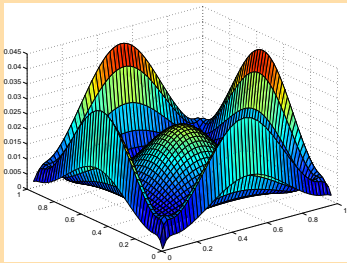
$$\frac{\|r\|}{\|b\|} = 0.7142$$

Adding 1 SGS step



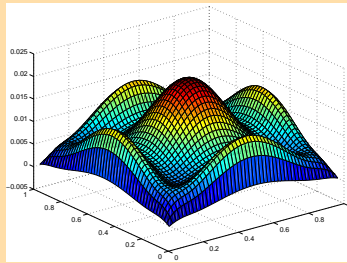
$$\frac{\|r\|}{\|b\|} = 0.0039$$

Adding 3 SGS steps



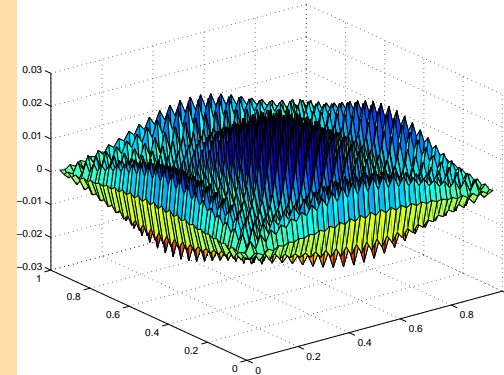
$$\frac{\|r\|}{\|b\|} = 0.0018$$

Adding 8 SGS steps



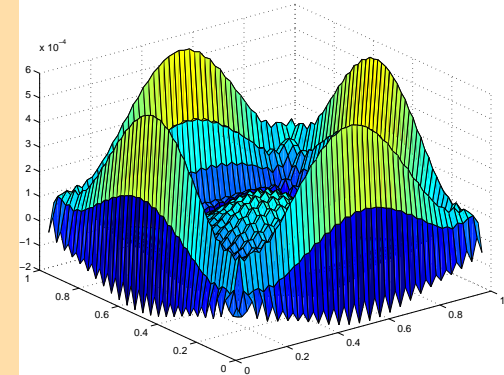
$$\frac{\|r\|}{\|b\|} = 0.0012$$

Adding now a CG correction



$$\frac{\|r\|}{\|r_{\text{previous}}\|} = 0.746$$

... and again 1 SGS step



$$\frac{\|r\|}{\|r_{\text{previous}}\|} = 0.0155$$

What we learned

For each coarse grid correction:

$$\mathbf{u} - \mathbf{u}^{(m+1)} = (I - p A_C^{-1} r A) (\mathbf{u} - \mathbf{u}^{(m)}) .$$

Cannot work alone because $\rho(I - p A_C^{-1} r A) \geq 1$.

For each smoothing step

$$\mathbf{u} - \mathbf{u}^{(m+1)} = (I - M^{-1} A) (\mathbf{u} - \mathbf{u}^{(m)}) .$$

Not efficient alone because $\rho(I - M^{-1} A) \approx 1$.

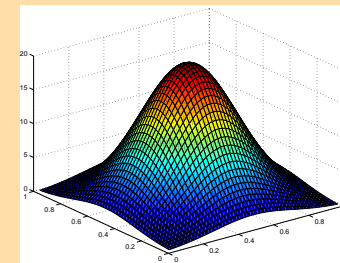
However

$$\rho\left((I - M^{-1} A) (I - p A_C^{-1} r A) (I - M^{-1} A) \right) \ll 1$$

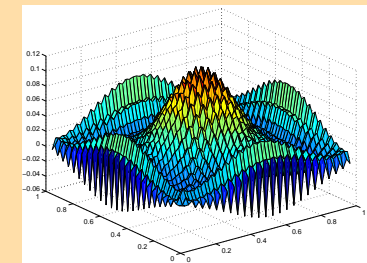
Rmk: if $A = A^T$, we assume $M = M^T$.

How it works

Initial residual

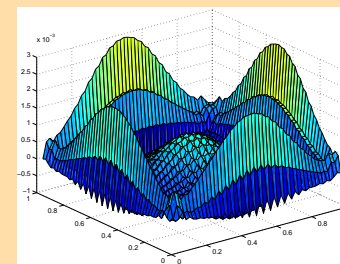


1 multigrid step



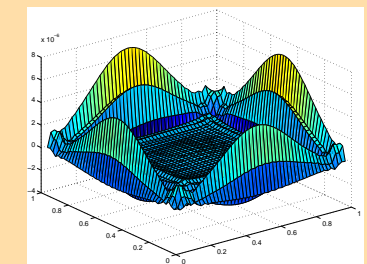
$$\frac{\|r\|}{\|b\|} = 3.9 \cdot 10^{-3}$$

2 multigrid steps



$$\frac{\|r\|}{\|b\|} = 7.7 \cdot 10^{-5}$$

4 multigrid steps



$$\frac{\|r\|}{\|b\|} = 1.7 \cdot 10^{-7}$$

Geometric multigrid

- Simple in its principles.
- Complicate to analyze.
- Not robust: simple ideas not always lead to efficient schemes.
- There is a lot of research works on multigrid applications.

Algebraic multigrid

- More user friendly (“black box”).
- More robust.
- ... sacrificing somewhat on efficiency.

Algebraic multigrid and multilevel methods – p.17/66

- Coarsening: F/C partitioning of the unknowns.
- Interpolation J_{FC} and prolongation $p = \begin{pmatrix} J_{FC} \\ I \end{pmatrix}$ satisfying $p e_C = e$.
- For the restriction, one often takes $r = \beta \begin{pmatrix} J_{FC}^T & I \end{pmatrix} = \beta p^T$ with β such that $r e = e_C$.
- For A_C one may rely on the Galerkin approximation:

$$A_C = r A p \quad \text{or} \quad \hat{A}_C = p^T A p$$

with coarse grid correction given by $p \hat{A}_C^{-1} p^T$.

- $I - p \hat{A}_C^{-1} p^T A = \left(I - p \hat{A}_C^{-1} p^T A \right)^2$ (projector)

Algebraic multigrid and multilevel methods – p.18/66

Two-grid AMG as a preconditioner

AMG as preconditioner

$\mathbf{v} = B_{\text{AMG}}^{-1} \mathbf{r}$ computed as

1. $\mathbf{t} = M^{-1} \mathbf{r}$; $\mathbf{w} = \mathbf{r} - A \mathbf{t}$
2. $\mathbf{y}_C = \mathbf{w}_C + J_{FC}^T \mathbf{w}_F$
3. Solve $\hat{A}_C \mathbf{z}_C = \mathbf{y}_C$
4. $\mathbf{z}_F = J_{FC} \mathbf{z}_C$
5. $\mathbf{v} = \mathbf{t} + \mathbf{z} + M^{-1} (\mathbf{w} - A \mathbf{z})$

$$I - B_{\text{AMG}}^{-1} A = (I - M^{-1} A) (I - p \hat{A}_C^{-1} p^T A) (I - M^{-1} A).$$

Algebraic multigrid and multilevel methods – p.19/66

Multilevel is not multigrid !

Coarse grid correction:

$$p \hat{A}_C^{-1} p^T \quad \text{with} \quad p = \begin{pmatrix} J_{FC} \\ I \end{pmatrix}.$$

Let's try an additive complement

$$q Q_{FF}^{-1} q^T \quad \text{with} \quad q = \begin{pmatrix} I \\ 0 \end{pmatrix}.$$

($Q_{FF} \approx q^T A q = A_{FF}$)

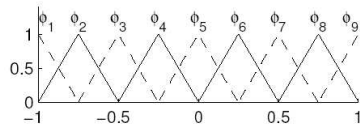
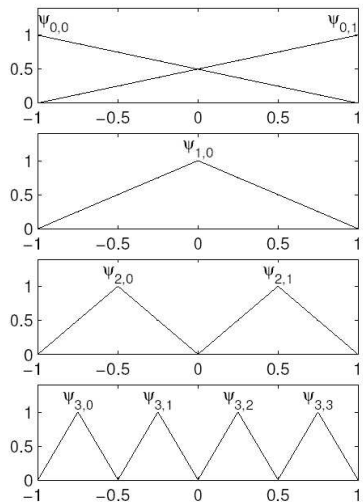
Corresponding preconditioner:

$$B_{\text{HBBD}}^{-1} = \begin{pmatrix} I & J_{FC} \\ & I \end{pmatrix} \begin{pmatrix} Q_{FF}^{-1} & \\ & \hat{A}_C^{-1} \end{pmatrix} \begin{pmatrix} I & \\ J_{FC}^T & I \end{pmatrix}.$$

Algebraic multigrid and multilevel methods – p.20/66

Linear Hierarchical Basis Functions

level 0: 2 functions
level 1: 1 function
level 2: 2 functions
level 3: 4 functions

 Standard Linear Basis Functions
9 functions


Finite element matrices are better conditioned whenever expressed in the hierarchical basis

Algebraic multigrid and multilevel methods – p.21/66

In finite element applications with regular refinement,

$$J = \begin{pmatrix} I & J_{FC} \\ & I \end{pmatrix}$$

performs the basis transformation $(hb_tl) \rightarrow (nb)$
 $((hb_tl)$: coarse nodal basis $(2h)$ + compl. functions (h)).

Matrix in this basis:

$$\begin{aligned} \hat{A} &= J^T A J = \begin{pmatrix} I & \\ J_{FC}^T & I \end{pmatrix} \begin{pmatrix} A_{FF} & A_{FC} \\ A_{CF} & A_{CC} \end{pmatrix} \begin{pmatrix} I & J_{FC} \\ & I \end{pmatrix} \\ &= \begin{pmatrix} A_{FF} & A_{FC} + A_{FF} J_{FC} \\ A_{CF} + J_{FC}^T A_{FF} & \hat{A}_C \end{pmatrix}. \end{aligned}$$

Algebraic multigrid and multilevel methods – p.22/66

Additive two-level

Matrix in (hb_tl) : $\hat{A} = J^T A J$.

Two-grid with additive complement:

$$B_{\text{HBB D}}^{-1} = \begin{pmatrix} I & J_{FC} \\ & I \end{pmatrix} \begin{pmatrix} Q_{FF}^{-1} & \\ & \hat{A}_C^{-1} \end{pmatrix} \begin{pmatrix} I & \\ J_{FC}^T & I \end{pmatrix} = J \hat{B}_{\text{HBB D}}^{-1} J^T,$$

where

$$\hat{B}_{\text{HBB D}} = \begin{pmatrix} Q_{FF} & \\ & \hat{A}_C \end{pmatrix} \approx \begin{pmatrix} A_{FF} & \\ & \hat{A}_C \end{pmatrix},$$

which is the block diagonal part of \hat{A} . Further,

$$B_{\text{HBB D}}^{-1} A = \left(J \hat{B}_{\text{HBB D}}^{-1} J^T \right) \left(J^{-T} \hat{A} J^{-1} \right) = J \hat{B}_{\text{HBB D}}^{-1} \hat{A} J^{-1}.$$

Algebraic multigrid and multilevel methods – p.23/66

The strengthened C.B.S. constant

We assume A symmetric and positive definite
Definition

$$\hat{\gamma} = \max_{\mathbf{v} = \begin{pmatrix} \mathbf{v}_F \\ 0 \end{pmatrix} \neq 0, \mathbf{w} = \begin{pmatrix} 0 \\ \mathbf{w}_C \end{pmatrix} \neq 0} \frac{|\mathbf{v}^T \hat{A} \mathbf{w}|}{\left(\mathbf{v}^T \hat{A} \mathbf{v} \right)^{1/2} \left(\mathbf{w}^T \hat{A} \mathbf{w} \right)^{1/2}}.$$

Property. If $\hat{A} = \sum_{\ell} \hat{A}_{\ell}$ and if, $\forall \ell$, $\hat{\gamma}_{\ell}$ is such that, for all $\mathbf{v} = \begin{pmatrix} \mathbf{v}_F \\ 0 \end{pmatrix}$, $\mathbf{w} = \begin{pmatrix} 0 \\ \mathbf{w}_C \end{pmatrix}$

$$|\mathbf{v}^T \hat{A}_{\ell} \mathbf{w}| \leq \hat{\gamma}_{\ell} \left(\mathbf{v}^T \hat{A}_{\ell} \mathbf{v} \right)^{1/2} \left(\mathbf{w}^T \hat{A}_{\ell} \mathbf{w} \right)^{1/2},$$

then: $\hat{\gamma} \leq \max_{\ell} \hat{\gamma}_{\ell}$

$\rightarrow \hat{\gamma}$ may often be bounded away from 1.

Algebraic multigrid and multilevel methods – p.24/66

Let

$$\hat{A} = \begin{pmatrix} A_{FF} & \hat{A}_{FC} \\ \hat{A}_{CF} & \hat{A}_C \end{pmatrix}, \quad \hat{D} = \begin{pmatrix} A_{FF} & \\ & \hat{A}_C \end{pmatrix},$$

One has

$$\kappa(\hat{D}^{-1}\hat{A}) = \frac{\lambda_{\max}(\hat{D}^{-1}\hat{A})}{\lambda_{\min}(\hat{D}^{-1}\hat{A})} = \frac{1 + \hat{\gamma}}{1 - \hat{\gamma}}.$$

Let $S_A = A_{CC} - A_{CF} A_{FF}^{-1} A_{FC} = \hat{A}_C - \hat{A}_{CF} A_{FF}^{-1} \hat{A}_{FC}$.

One has

$$\lambda_{\min}(\hat{A}_C^{-1} S_A) = 1 - \hat{\gamma}^2,$$

$$\lambda_{\max}(\hat{A}_C^{-1} S_A) \leq 1.$$

Preconditioning by HBBD

$\mathbf{v} = B_{\text{HBBD}}^{-1} \mathbf{r}$ computed as

1. $\mathbf{y}_F = Q_{FF}^{-1} \mathbf{r}_F$
2. $\mathbf{y}_C = \mathbf{r}_C + J_{FC}^T \mathbf{r}_F$
3. Solve $\hat{A}_C \mathbf{v}_C = \mathbf{y}_C$
4. $\mathbf{z}_F = J_{FC} \mathbf{v}_C$
5. $\mathbf{v}_F = \mathbf{z}_F + \mathbf{y}_F$

$$\kappa \approx \frac{1 + \hat{\gamma}}{1 - \hat{\gamma}}$$

$$B_{\text{HBBD}}^{-1} = \begin{pmatrix} I & J_{FC} \\ & I \end{pmatrix} \begin{pmatrix} Q_{FF}^{-1} & \\ & \hat{A}_C^{-1} \end{pmatrix} \begin{pmatrix} I & \\ J_{FC}^T & I \end{pmatrix} \\ = q Q_{FF}^{-1} q^T + p \hat{A}_C^{-1} p^T.$$

Two-level block factorization

$$\hat{A} = \begin{pmatrix} A_{FF} & \hat{A}_{FC} \\ \hat{A}_{CF} & \hat{A}_C \end{pmatrix} \\ = \begin{pmatrix} I & & \\ \hat{A}_{CF} A_{FF}^{-1} & I & \\ & & I \end{pmatrix} \begin{pmatrix} A_{FF} & \\ & S_A \end{pmatrix} \begin{pmatrix} I & A_{FF}^{-1} \hat{A}_{FC} \\ & I \end{pmatrix} \\ \approx \begin{pmatrix} I & & \\ \hat{A}_{CF} Q_{FF}^{-1} & I & \\ & & I \end{pmatrix} \begin{pmatrix} Q_{FF} & \\ & \hat{A}_C \end{pmatrix} \begin{pmatrix} I & Q_{FF}^{-1} \hat{A}_{FC} \\ & I \end{pmatrix} \\ = \hat{B}_{\text{HBBF}}$$

$$B_{\text{HBBF}}^{-1} = J \hat{B}_{\text{HBBF}}^{-1} J^T$$

Two-level block factorization (cont.)

Preconditioning by HBBF

$\mathbf{v} = B_{\text{HBBF}}^{-1} \mathbf{r}$ computed as

1. $\mathbf{y}_F = Q_{FF}^{-1} \mathbf{r}_F$
2. $\mathbf{y}_C = \mathbf{r}_C - A_{CF} \mathbf{y}_F + J_{FC}^T (\mathbf{r}_F - A_{FF} \mathbf{y}_F)$
3. Solve $\hat{A}_C \mathbf{v}_C = \mathbf{y}_C$
4. $\mathbf{z}_F = J_{FC} \mathbf{v}_C$
5. $\mathbf{v}_F = \mathbf{z}_F + Q_{FF}^{-1} (\mathbf{r}_F - A_{FC} \mathbf{v}_C - A_{FF} \mathbf{z}_F)$

$$\kappa \approx \frac{1}{1 - \hat{\gamma}^2}$$

$$B_{\text{HBBF}}^{-1} = J \begin{pmatrix} I & -Q_{FF}^{-1} \hat{A}_{FC} \\ & I \end{pmatrix} \begin{pmatrix} Q_{FF}^{-1} & \\ & \hat{A}_C^{-1} \end{pmatrix} \begin{pmatrix} I & -Q_{FF}^{-1} \hat{A}_{FC} \\ & I \end{pmatrix} J^T$$

Preconditioning by HBMG

$\mathbf{v} = B_{\text{HBMG}}^{-1} \mathbf{r}$ computed as

1. $\mathbf{y}_F = Q_{FF}^{-1} \mathbf{r}_F$
2. $\mathbf{y}_C = \mathbf{r}_C - A_{CF} \mathbf{y}_F + J_{FC}^T (\mathbf{r}_F - A_{FF} \mathbf{y}_F)$
3. Solve $\hat{A}_C \mathbf{v}_C = \mathbf{y}_C$
4. $\mathbf{z}_F = J_{FC} \mathbf{v}_C + \mathbf{y}_F$
5. $\mathbf{v}_F = \mathbf{z}_F + Q_{FF}^{-1} (\mathbf{r}_F - A_{FC} \mathbf{v}_C - A_{FF} \mathbf{z}_F)$

$$\kappa \approx \frac{1}{1 - \hat{\gamma}^2}$$

$$B_{\text{HBMG}}^{-1} = J \begin{pmatrix} I & -Q_{FF}^{-1} \hat{A}_{FC} \\ & I \end{pmatrix} \begin{pmatrix} 2Q_{FF}^{-1} - Q_{FF}^{-1} A_{FF} Q_{FF}^{-1} & \\ & \hat{A}_C^{-1} \end{pmatrix} \dots$$

Algebraic multigrid and multilevel methods – p.29/66

Elementary algebra yields

$$I - B_{\text{HBMG}}^{-1} A = (I - RA)(I - p \hat{A}_C^{-1} p^T A)(I - RA)$$

with

$$R = \begin{pmatrix} Q_{FF}^{-1} & 0 \\ 0 & 0 \end{pmatrix}.$$

Reminder:

$$I - B_{\text{AMG}}^{-1} A = (I - M^{-1} A)(I - p \hat{A}_C^{-1} p^T A)(I - M^{-1} A).$$

Algebraic multigrid and multilevel methods – p.30/66

Block factorization without h.b.

$$\begin{aligned} A &= \begin{pmatrix} A_{FF} & A_{FC} \\ A_{CF} & A_{CC} \end{pmatrix} \\ &= \begin{pmatrix} I & & \\ A_{CF} A_{FF}^{-1} & I & \end{pmatrix} \begin{pmatrix} A_{FF} & \\ & S_A \end{pmatrix} \begin{pmatrix} I & A_{FF}^{-1} A_{FC} \\ & I \end{pmatrix} \\ &\approx \begin{pmatrix} I & & \\ A_{CF} P_{FF}^{-1} & I & \end{pmatrix} \begin{pmatrix} P_{FF} & \\ & \hat{A}_C \end{pmatrix} \begin{pmatrix} I & P_{FF}^{-1} A_{FC} \\ & I \end{pmatrix} \\ &= B_{\text{MBF}} \end{aligned}$$

Algebraic multigrid and multilevel methods – p.31/66

Block factorization without h.b. (cont.)

Preconditioning by MBF

$\mathbf{v} = B_{\text{MBF}}^{-1} \mathbf{r}$ computed as

1. $\mathbf{y}_F = P_{FF}^{-1} \mathbf{r}_F$
2. $\mathbf{y}_C = \mathbf{r}_C - A_{CF} \mathbf{y}_F$
3. Solve $\hat{A}_C \mathbf{v}_C = \mathbf{y}_C$
4. $\mathbf{v}_F = P_{FF}^{-1} (\mathbf{r}_F - A_{FC} \mathbf{v}_C)$

$$\kappa \approx \frac{1}{1 - \hat{\gamma}^2}$$

Possibly unstable!

$$B_{\text{MBF}}^{-1} = \begin{pmatrix} I & -P_{FF}^{-1} A_{FC} \\ & I \end{pmatrix} \begin{pmatrix} P_{FF}^{-1} & \\ & \hat{A}_C^{-1} \end{pmatrix} \begin{pmatrix} I & \\ -A_{CF} P_{FF}^{-1} & I \end{pmatrix}$$

Algebraic multigrid and multilevel methods – p.32/66

$$B_{\text{HBB D}}^{-1} = q Q_{\text{FF}}^{-1} q^T + p \hat{A}_C^{-1} p^T .$$

$$B_{\text{HBB F}}^{-1} = q Q_{\text{FF}} q^T + \tilde{p} \hat{A}_C^{-1} \tilde{p}^T ,$$

$$B_{\text{HBM G}}^{-1} = q (Q_{\text{FF}}^{-1} + Q_{\text{FF}}^{-T} - Q_{\text{FF}}^{-1} A_{\text{FF}} Q_{\text{FF}}^{-T}) q^T + \tilde{p} \hat{A}_C^{-1} \tilde{p}^T ,$$

$$\tilde{p} = \begin{pmatrix} -Q_{\text{FF}}^{-1} A_{\text{FC}} + (I - Q_{\text{FF}}^{-1} A_{\text{FF}}) J_{\text{FC}} \\ I \end{pmatrix} .$$

$$B_{\text{MBF}}^{-1} = q P_{\text{FF}}^{-1} q^T + \bar{p} \hat{A}_C^{-1} \bar{p}^T ,$$

$$\bar{p} = \begin{pmatrix} -P_{\text{FF}}^{-1} A_{\text{FC}} \\ I \end{pmatrix} .$$

\bar{p} has to define a “correct” interpolation.

Algebraic multigrid and multilevel methods – p.33/66

Correct interpolation

- Essential requirement:

$$\bar{p} = \begin{pmatrix} -P_{\text{FF}}^{-1} A_{\text{FC}} \\ I \end{pmatrix} \text{ good for low energy modes,}$$

i.e. vectors \mathbf{v} such that $A\mathbf{v} \approx 0$.

- Scalar elliptic PDEs: one such vector:

$$\mathbf{e} = (1 \ \dots \ 1)^T .$$

- If one satisfies the row-sum criterion

$$P_{\text{FF}} \mathbf{e}_F = A_{\text{FF}} \mathbf{e}_F ,$$

then

$$A_{\text{FF}} \mathbf{e}_F + A_{\text{FC}} \mathbf{e}_C \approx 0 \Rightarrow P_{\text{FF}}^{-1} A_{\text{FC}} \mathbf{e}_C \approx \mathbf{e}_F .$$

Algebraic multigrid and multilevel methods – p.35/66

h^{-1}	$P_{\text{FF}}^{-1} A_{\text{FF}}$			$B_{\text{MBF}}^{-1} A$		
	λ_{\min}	λ_{\max}	κ	λ_{\min}	λ_{\max}	κ
MILU preconditioning of A_{FF}						
16	1.00	1.20	1.20	0.51	1.25	2.45
32	1.00	1.21	1.21	0.50	1.27	2.54
64	1.00	1.21	1.21	0.50	1.29	2.58
128	1.00	1.21	1.21	0.50	1.29	2.58
ILU preconditioning of A_{FF}						
16	0.88	1.09	1.24	0.510	1.42	2.78
32	0.88	1.09	1.24	0.380	2.28	6.00
64	0.87	1.09	1.25	0.176	4.97	28.30
128	0.87	1.09	1.25	0.058	15.00	258.00

Algebraic multigrid and multilevel methods – p.34/66

Correct interpolation (cont.)

If A is a symmetric M-matrix with nonnegative row-sum (SPD with nonpositive offdiagonal entries), several results available.

For instance:

$$\bar{A} = \bar{J}^T A \bar{J}$$

with

$$\bar{J} = \begin{pmatrix} I & -P_{\text{FF}}^{-1} A_{\text{FC}} \\ & I \end{pmatrix}$$

satisfies

$$\bar{\gamma} \leq \sqrt{1 - \frac{1}{\kappa(P_{\text{FF}}^{-1} A_{\text{FF}})}} .$$

Algebraic multigrid and multilevel methods – p.36/66

HBBF, HBMG: $\kappa \approx \frac{1}{1 - \hat{\gamma}^2}$; MBF: $\kappa \approx \frac{1}{1 - \hat{\gamma}^2}$ (!).

AMG

Assumption: $2M - A$ SPD or, equivalently, $\rho(I - M^{-1}A) < 1$. One has

$$\kappa(B_{\text{AMG}}^{-1}A) \leq \mu$$

where

$$\mu = \max_{\mathbf{z} \neq 0} \frac{(\mathbf{z}_F - J_{FC}\mathbf{z}_C)^T X_{FF} (\mathbf{z}_F - J_{FC}\mathbf{z}_C)}{\mathbf{z}^T A \mathbf{z}},$$

with X_{FF} being the top left block of

$$X = M(2M - A)^{-1}M.$$

Further,

$$\begin{aligned} \frac{1}{1 - \hat{\gamma}^2} \leq \mu &\leq \frac{1}{\lambda_{\min}(X_{FF}^{-1}A_{FF})} \frac{1}{1 - \hat{\gamma}^2} \\ &\leq \frac{1}{\lambda_{\min}(M_{FF}^{-1}A_{FF}) (2 - \lambda_{\max}(M^{-1}A))} \frac{1}{1 - \hat{\gamma}^2}. \end{aligned}$$

Example: SGS smoothing: $\lambda_{\max}(M^{-1}A) = 1$.

$$\frac{1}{1 - \hat{\gamma}^2} \leq \mu \leq \frac{1}{\lambda_{\min}(M_{FF}^{-1}A_{FF})} \frac{1}{1 - \hat{\gamma}^2}.$$

Algebraic analysis of AMG (cont.)

Quality of the interpolation measured with

$$\tau = \max_{\mathbf{z} \neq 0} \frac{(\mathbf{z}_F - J_{FC}\mathbf{z}_C)^T D_{FF} (\mathbf{z}_F - J_{FC}\mathbf{z}_C)}{\mathbf{z}^T A \mathbf{z}}$$

($D_{FF} = \text{diag}(A_{FF})$).

There holds

$$\tau \leq \frac{1}{\lambda_{\min}(D_{FF}^{-1}A_{FF})} \frac{1}{1 - \hat{\gamma}^2}$$

$$\tau \geq \max \left(\frac{1}{\lambda_{\max}(D_{FF}^{-1}A_{FF})} \frac{1}{1 - \hat{\gamma}^2}, \frac{1}{\lambda_{\min}(D_{FF}^{-1}A_{FF})} \right).$$

What we learned

- All methods work or fail together.
- They are relatively equivalent with respect to algebraic analysis (except “additive” HBBF).
- However they mimic “geometric” methods that behave differently in a multigrid or multilevel context.
- The F/C partitioning has to be such that A_{FF} is well conditioned.
- The interpolation J_{FC} has to be such that $\hat{\gamma}$ is away from 1.
- MBF needs special care; it does not require explicitly J_{FC} , but \hat{A}_C needs to be provided.

Consider

$$\begin{pmatrix} I & \\ -A_{CF}A_{FF}^{-1} & I \end{pmatrix} \begin{pmatrix} A_{FF} & A_{FC} \\ A_{CF} & A_{CC} \end{pmatrix} \begin{pmatrix} I & -A_{FF}^{-1}A_{FC} \\ & I \end{pmatrix} = \begin{pmatrix} A_{FF} & \\ & S_A \end{pmatrix}.$$

Block diagonal $\rightarrow \gamma = 0$.

$\rightarrow -A_{FF}^{-1}A_{FC}$ is the ideal algebraic interpolation.

However: $\hat{A}_C = \begin{pmatrix} J_{FC}^T & I \end{pmatrix} A \begin{pmatrix} J_{FC} \\ I \end{pmatrix}$

$\rightarrow J_{FC}$ has to remain sparse.

Direct interpolation in AMG for **M-matrices** with nonnegative row-sum:

$$(J_{FC})_{ij} = \begin{cases} \frac{-\sum_{j \neq i} |(A)_{ij}|}{(A_{FF})_{ii}} \frac{(A_{FC})_{ij}}{\sum_{j \in C} |(A_{FC})_{ij}|} & \text{if } a_{ij} \text{ "strong"} \\ 0 & \text{if } a_{ij} \text{ "weak"}. \end{cases}$$

Property:

$$\tau \leq \max_{i \in F} \frac{\sum_{j \neq i} |(A)_{ij}|}{\sum_{j \in C} |(A_{FC})_{ij}|}$$

(Reminder: $\tau \approx \frac{1}{1-\tilde{\gamma}^2}$).

- Possible improvement: take also into account “indirect” couplings (J_{FC} less sparse).
- Essentially positive-type matrices with nonnegative row-sum: split $A = A_M + A_P$ where $\text{offdiag}(A_P) = \max(O, \text{offdiag}(A))$ and $A_P e = 0$; apply previous scheme to A_M ; the bound on τ depends now on $\kappa(A_M^{-1}A)$.
- General case: no obvious solution so far if A is not (weakly) diagonally dominant.

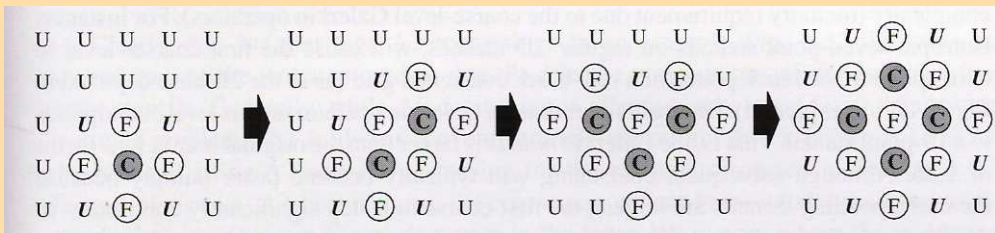
Standard coarsening in AMG

First classify the negative couplings in **strong** and **weak**, according some given threshold.

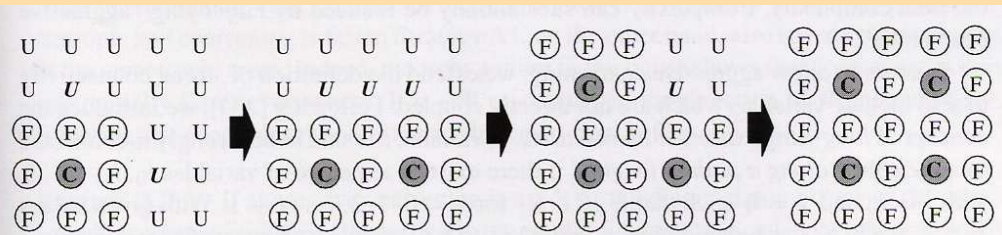
Next, repeat, till all nodes are marked either **coarse** or **fine**:

1. select an unmarked node as next coarse grid node, according to some priority rule (designed so as to favor a regular covering of the matrix graph);
2. select as fine grid nodes all nodes strongly negative coupled to this new coarse grid node.

Five-point stencil



Nine-point stencil



- Each F node is strongly negative coupled to at least 1 C node
 → standard interpolation works.
- Slow coarsening in case of low connectivity, anisotropy or strong asymmetry. (Too fast coarsening in case of high connectivity).
- May be cured with **aggressive coarsening**. Requires specialized interpolation.
- The number of nonzero entries per row tends to grow from level to level.
- May be sensitive to the Strong/Weak coupling threshold.
- All in all, works reasonably in many cases.

Aggregation

- Group nodes into **aggregates** G_i (partitioning of $[1, n]$).
- (Possible) prolongation p :

$$(p)_{ij} = \begin{cases} 1 & \text{if } i \in G_j \\ 0 & \text{otherwise} \end{cases} .$$

- Coarse grid matrix: $\hat{A}_C = p^T A p$ given by

$$(\hat{A}_C)_{ij} = \sum_{k \in G_i} \sum_{\ell \in G_j} a_{k\ell} .$$

- Optionally select a C node in each aggregates; other nodes are then F nodes. Associated interpolation:

$$\forall i \in F, j \in C : (J_{FC})_{ij} = \begin{cases} 1 & \text{if } i \in G_j \\ 0 & \text{otherwise} \end{cases} .$$

Example: pairwise aggregation

Definition: $S_i = \{ j \neq i \mid a_{ij} < -\beta \max_{a_{ik} < 0} |a_{ik}| \}$

Initialization: $F = \emptyset; C = \emptyset; U = [1, n];$

For all i : $m_i = |\{ j \in U \mid i \in S_j \}|$.

Algorithm: While $U \neq \emptyset$ do

- select $i \in U$ with minimal m_i
- select $j \in U$ such that $a_{ij} = \min_{k \in U} a_{ik}$
- if $j \in S_i$:
 - $C = C \cup \{j\}, F = F \cup \{i\}, G_j = \{i, j\}, U = U \setminus \{i, j\}$
 - update: $m_k = m_k - 1$ for $k \in S_i$ and $k \in S_j$

otherwise:

- $C = C \cup \{i\}, G_i = \{i\}, U = U \setminus \{i\}$
- update: $m_k = m_k - 1$ for $k \in S_i$

Algorithm:

1. Apply simple pairwise aggregation to A .
Output: (F_1, C_1) , and $G_i^{(1)}$, $i \in C_1$.
2. Compute the auxiliary matrix $A_1 = (a_{ij}^{(1)})$, $i, j \in C_1$
with

$$a_{ij}^{(1)} = \sum_{k \in G_i^{(1)}} \sum_{l \in G_j^{(1)}} a_{kl}.$$
3. Apply simple pairwise aggregation to A_1 .
Output: (F_2, C_2) , and $G_i^{(2)}$, $i \in C_2$.
4. $C = C_2$, $F = F_1 \cup F_2$, $G_i = \cup_{j \in G_i^{(2)}} G_j^{(1)}$, $i \in C$.

2D problem with anisotropy & discontinuity

Five-point finite difference approx. (uniform mesh) of

$$-a_x \frac{\partial^2 u}{\partial x^2} - a_y \frac{\partial^2 u}{\partial y^2} = f \quad \text{in } \Omega = (0, 1) \times (0, 1)$$

$$\begin{cases} u = 0 & \text{on } y = 1, 0 \leq x \leq 1 \\ \frac{\partial u}{\partial n} = 0 & \text{elsewhere on } \partial\Omega \end{cases}$$

$$\begin{cases} a_x = d, a_y = 1, f = 0 & \text{in } (0.65, 0.95) \times (0.05, 0.65) \\ a_x = 1, a_y = d, f = 0 & \text{in } (0.25, 0.45) \times (0.25, 0.45) \\ a_x = d, a_y = d, f = 1 & \text{in } (0.05, 0.25) \times (0.65, 0.95) \\ a_x = 1, a_y = 1, f = 0 & \text{elsewhere,} \end{cases}$$

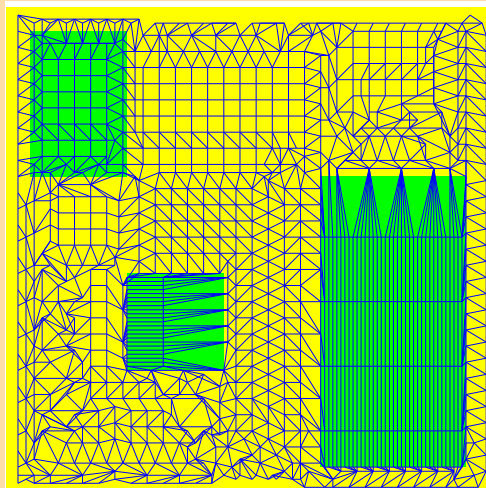
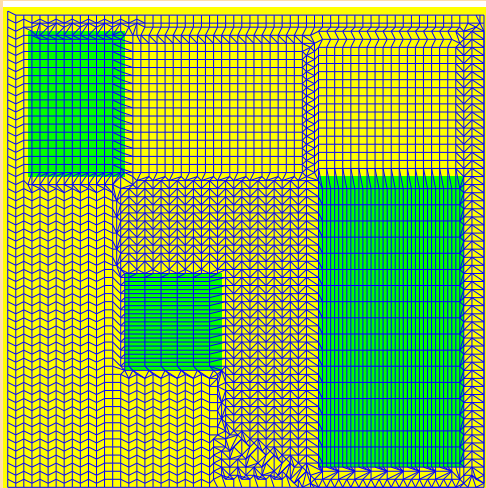
where d is a parameter.

Example ($d = 100$)

Double pairwise aggregation

First coarse grid

Second coarse grid



$$n_c = 3794, \frac{n}{n_c} = 3.83, \frac{nz}{n_c} = 5.46$$

$$n_c = 1025, \frac{n}{n_c} = 14.2, \frac{nz}{n_c} = 6.02$$

Some remarks

Geometric multigrid does not benefit from semi-coarsening

- A_{FF} may be badly conditioned
- has to be compensated by specialized smoothers.

Geometric schemes fix the coarsening and the interpolation; the smoother (the approximation to A_{FF}) is adapted to the problem.

Algebraic schemes fix the smoother (the approximation to A_{FF}); the coarsening is adapted to the problem.

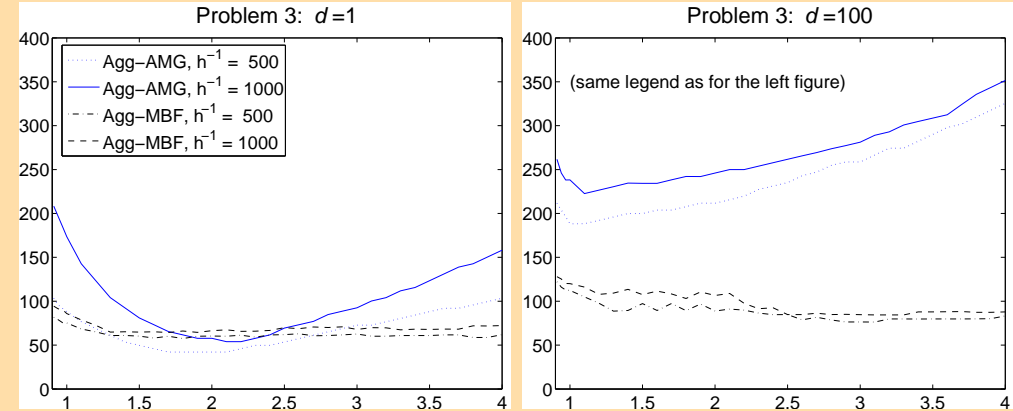
With algebraic schemes, the adaptation is automatic.

- Control of the coarsening speed.
- Insensitive to the Strong/Weak coupling threshold.
- Maintain the sparsity in coarse grid matrices, that are nevertheless “reasonable”, up to some scaling factor.
- The interpolation that is naturally associated with aggregation is bad (not an issue for MBF-based methods).
- Smoothed aggregation: optionally sparsify A into \tilde{A} , in such a way that $Ae = \tilde{A}e$; then:

$$p_{sm\ agg} = (I - \omega \tilde{D}^{-1} \tilde{A}) p_{agg}$$

where $\tilde{D} = \text{diag}(\tilde{A})$.

Performance of AMG and MBF with aggregation



Relative solution cost – vs – scaling of the coarse grids

Checking the F/C partitioning

A_{FF} has to be well conditioned. This may be a posteriori checked.

Compatible relaxation (AMG)

Perform smoothing on a random r.h.s while freezing the values at C variables. If the error at F variables does not decay quickly, adapt the partitioning by moving to C some of the slowly convergent F variables.

Remark

Amounts to check the conditioning of $M_{FF}^{-1}A_{FF}$.

Remember that

$$\kappa_{AMG} \sim (\lambda_{\min}(M_{FF}^{-1}A_{FF}) (2 - \lambda_{\max}(M^{-1}A)))^{-1}$$

Checking the F/C partitioning (cont.)

Dynamic MILU

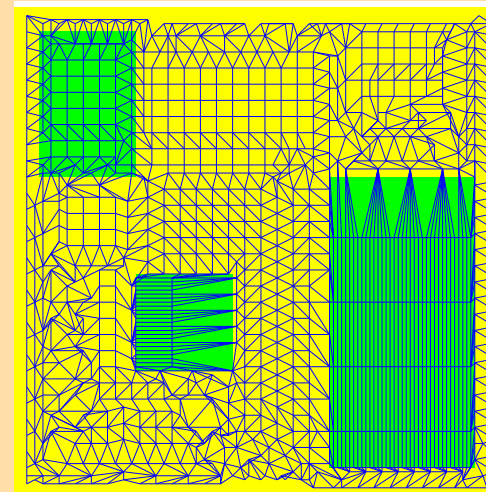
The size of the pivots in a modified ILU ($P_{FF}e_F = A_{FF}e_F$) factorization is a good indication of the conditioning.

For instance, in some cases, letting $P_{FF} = L_{FF} Q_{FF}^{-1} U_{FF}$ with $\text{diag}(L_{FF}) = \text{diag}(U_{FF}) = Q_{FF}$, if $Q_{FF} \geq \xi \text{diag}(A_{FF})$ for some $\xi > \frac{1}{2}$, then

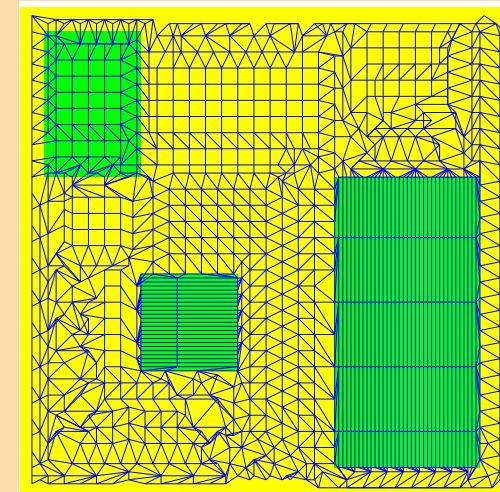
$$\kappa(P_{FF}^{-1}A_{FF}) \leq \frac{1}{2 - \xi^{-1}}$$

1. *Repeat=False*.
2. (re)initialize:
 $Q_{FF} = \text{diag}(A_{FF})$, $L_{FF} = \text{lower}(A_{FF})$,
 $U_{FF} = \text{upper}(A_{FF})$.
3. for $k = 1, \dots, n$, $k \in F$:
 if $q_{kk} \geq \gamma a_{kk}$:
 eliminate row & column k in A_{FF} according to the MILU algorithm
 otherwise: $F = F \setminus \{k\}$, $C = C \cup \{k\}$;
 Repeat=True.
4. If (*Repeat*), GoTo 1, possibly decreasing the value of γ .

Double pairwise aggregation, second coarse grid
 Without dynamic MILU With dynamic MILU



$$n_c = 1025, \quad \frac{n}{n_c} = 14.2, \quad \frac{nz}{n_c} = 6.02$$



$$n_c = 1124, \quad \frac{n}{n_c} = 12.9, \quad \frac{nz}{n_c} = 5.94$$

From two- to multi-level

- Exploit recursively the same ideas.
- Succession of grids (levels), each with its own F/C partitioning and interpolation J_{FC} , and also with its “ideal” preconditioner in which the matrix at the coarser level is inverted exactly.
- At some point the coarse grid matrix is indeed small enough to be factorized exactly.
- At every other level, the “ideal” preconditioner is adapted, exchanging the exact solution to $\hat{A}_C \mathbf{v}_C = \mathbf{y}_C$ for an approximate solution.
- Approximate $\hat{A}_C \mathbf{v}_C = \mathbf{y}_C$ with
 1 application of the preconditioner: V cycle.
 inner iterations: W cycle.

From two- to multi-level (cont.)

W cycles may be based on fixed point iterations, but Krylov (CG, GMRES) is more robust. Then:

- Except at the coarsest level, the so defined preconditioner is slightly variable from step to step
 → Flexible Krylov subspace methods (FCG, FGMRES).
- Inner iterations are exited when the relative residual error is less than 0.35, or when the number of iterations reaches $\text{int}[nz(A)/nz(\hat{A}_C)]$.

AMG: often efficient with V cycle

→ simplicity, consistency with slow coarsening.

The use of V cycle is based on experiment and mimicry of geometric schemes

→ it may be not robust to **rely** on V cycle.

Block factorization methods: require W cycle (geometric schemes do require it too)

→ need coarsening fast enough.

MBF with aggregation & dynamic MILU

“sol” = $\frac{\text{Cost of resolution}}{\text{Cost of 1 unprec. CG iter.}} \approx 28$ for geom. multigrid (on model problems).

d	$h^{-1} = 600$				$h^{-1} = 1200$			
	$\frac{n}{n_c}$	inner	iter.	sol.	$\frac{n}{n_c}$	inner	iter.	sol.
1	3.99	1.76	21	67.1	4.00	1.76	21	68.8
2	3.97	2.00	20	69.8	3.99	2.00	23	82.9
4	3.96	2.04	24	84.1	3.98	2.00	25	90.3
10	3.95	2.04	24	84.2	3.98	2.04	26	94.0
10^2	3.95	2.04	24	82.5	3.98	2.00	26	90.6
10^4	3.95	1.96	26	88.0	3.98	2.04	27	95.6
10^6	3.95	2.15	26	92.2	3.98	2.00	31	107.9

A non self-adjoint 3D problem

Seven-point FD approx. (upwind scheme) of

$$-\nu \Delta u + \bar{v} \nabla u = 0 \quad \text{in } \Omega = (0, 1) \times (0, 1) \times (0, 1)$$

$$\begin{cases} u = 1 & \text{on } z = 1, 0 \leq x, y \leq 1 \\ u = 0 & \text{elsewhere on } \partial\Omega \end{cases}$$

$$\bar{v}(x, y, z) = \begin{pmatrix} 2x(1-x)(2y-1)z \\ -(2x-1)y(1-y) \\ -(2x-1)(2y-1)z(1-z) \end{pmatrix};$$

$\nu = \infty$ corresponds to the Laplace equation.

Uniform mesh with constant mesh size h .

Stretched mesh: refined in such a way that the ratio of maximum mesh size to minimum mesh size is equal to 200, the ratio of subsequent mesh sizes being constant.

3D problem: numerical results

ν	101 × 101 × 101 grid				201 × 201 × 201 grid			
	$\frac{n}{n_c}$	inner	iter.	sol.	$\frac{n}{n_c}$	inner	iter.	sol.
∞	4.00	2.00	15	79.6	4.00	2.00	15	82.0
1	3.76	2.00	17	109.7	3.81	2.00	17	108.3
10^{-2}	3.75	2.00	18	119.7	3.84	1.94	18	111.7
10^{-4}	3.93	2.00	21	136.9	3.93	2.00	21	136.8
10^{-6}	3.93	2.00	26	167.2	3.96	2.00	30	203.0
∞	3.91	1.94	16	79.3	3.94	1.88	17	85.8
1	3.91	1.94	16	80.0	3.94	1.88	17	85.8
10^{-2}	3.92	1.65	20	95.5	3.95	1.94	17	85.3
10^{-4}	3.46	1.81	21	117.1	3.64	1.87	23	125.8
10^{-6}	3.64	2.00	27	171.0	3.28	2.00	27	186.3

Many textbooks on multigrid, but few address algebraic schemes.

- U. Trottenberg, C.W. Oosterlee, and A. Schüller. *Multigrid*. Academic Press, London, 2001.

is recommended for a general introduction to multigrid; it contains in appendix the best available review on AMG:

- K. Stüben. *An Introduction to Algebraic Multigrid*. In Trottenberg et al., 2001. Appendix A.

Other results in research papers. Let mention mine !

- Algebraic multigrid and algebraic multilevel methods: a theoretical comparison
- Aggregation-based algebraic multilevel preconditioning (see homepage for details and download)

Algebraic multigrid and multilevel methods – p.65/66

PhD Fellowship

Area: numerical nuclear reactor simulation

Collaboration between ULB and Framatome ANP

Location: Framatome ANP GmbH in Erlangen, Germany (main European research center of the group) with periodical stays in Brussels.

Task: adaptation of advanced preconditioned iterative techniques to nuclear reactor simulation.

Please contact me for further information.

ynotay@ulb.ac.be

Algebraic multigrid and multilevel methods – p.66/66